Musikverwaltung

Die Standard-Musikapps auf Smartphones und Tablets unterstützen mit ihren Funktionen das Verwalten von Musiksammlungen mit Unterhaltungsmusik. Künstlern (Einzelpersonen oder Musikgruppen mit eindeutigem Namen) sind einzelne Alben zugeordnet. Ein Album enthält mehrere Musikstücke (Titel) in einer bestimmten Reihenfolge. Jeder Titel wird für genau ein Album aufgenommen. Nicht berücksichtigt werden sollen hier Sampler (Alben mit Titeln von verschiedenen Künstlern), gemeinsame Projekte (einzelne Titel von Alben mit mehreren Künstlern) und Best-Of-Alben (Alben mit Titeln, die schon auf anderen Alben enthalten sind).

Man kann die eigene Sammlung nach Künstlernamen, Albennamen oder Titelnamen sortieren lassen oder durchsuchen. Zudem lassen sich weitere Informationen abrufen, wie z. B. das Erscheinungsdatum eines Albums.

Außerdem ist das Anlegen von Playlists möglich. Playlists haben einen frei wählbaren Namen. Sie sind eine Zusammenstellung einzelner Titel (auch von verschiedenen Alben) in einer bestimmten Reihenfolge.

Beispiele: *Michael Jackson* und *Queen* sind Beispiele für Künstler. Alben von Queen sind *A Night at the Opera* und *A Kind of Magic*. Das Album *A Night at the Opera* enthält die Titel 1) *Death On Two Legs,* 2) *Lazing On a Sunday Afternoon, ...*

Aufgaben

1.1 Modellieren Sie für die beschriebene Situation ein vollständig beschriftetes ER-Diagramm mit Angabe der Kardinalitäten und Optionalitäten.

(4 BE)

1.2 Erläutern Sie den Beziehungstyp zwischen den Entitätstypen *Titel* und *Playlist*.

(2 BE)

1.3 Material 1 zeigt eine Relation, mithilfe derer Playlists gespeichert werden können. Begründen Sie, in welcher Normalform sich die Relation befindet. Überführen Sie diese inklusive der enthaltenen Daten schrittweise in die 3. Normalform.

(4 BE)

2 Erläutern Sie den Aufbau und die inhaltliche Bedeutung der folgenden SQL-Anweisungen:

2.1
```
SELECT Album.Name, Titel.Name
FROM Album JOIN Titel ON Album.ID = Titel.AlbumID
WHERE Titel.Name LIKE '%love%'
  AND Erscheinungsdatum BETWEEN '1980-01-01' AND '1989-12-31'
```

(3 BE)

2.2 SELECT Künstlername, COUNT(*) AS Albenanzahl
FROM Album
GROUP BY Künstlername
HAVING Albenanzahl > 20
ORDER BY Albenanzahl DESC

(2 BE)

3 Neben Sammlungen mit Unterhaltungsmusik wollen viele Nutzer auch Sammlungen von Hörbüchern, Hörspielen oder ernster Musik verwalten. Die Standard-Musikapps bieten die hierbei benötigten Funktionen häufig nicht an. Diese sind:

- eigene Zusammenstellungen von Titeln und Alben erzeugen
- für jede Zusammenstellung und jedes Album und jeden Titel einer Zusammenstellung die zuletzt gehörte Position merken
- Ausblendlisten anlegen, die in einer eigenen Zusammenstellung für ein Album verhindern, dass bestimmte Titelnummern abgespielt werden
- einen alternativen Startpunkt für Titel definieren, so dass diese nicht von Beginn an abgespielt werden

Material 2 zeigt das ER-Diagramm der Datenbank einer App, die sogenannte Boxen nutzt, um (Musik-)Sammlungen zu strukturieren. Dabei werden für Alben und Titel nur Adressen (URLs) gespeichert, unter denen diese verfügbar sind. Über diese Adressen sind Informationen wie der Name eines Albums abrufbar. Beziehungen zwischen Alben und Titeln werden von dieser App nicht verwaltet.

3.1 Erläutern Sie auf Basis des ER-Diagramms, wie die App die im Aufgabentext genannten Funktionen umsetzt.

(2 BE)

3.2 Überführen Sie das ER-Diagramm (Material 2) in ein optimiertes Relationenmodell.

(4 BE)

4 Implementieren Sie SQL-Anweisungen für die folgenden Anforderungen:

4.1 Es sollen die URLs aller Items der Box mit der *BoxID* 23, aufsteigend sortiert nach dem Attribut *Position* ausgegeben werden.

(2 BE)

4.2 Es soll die Tabelle *Ausblendtitelnummern* mit allen Attributen angelegt und der Primär- und Fremdschlüssel festgelegt werden.

(2 BE)

4.3 Der Benutzer schiebt das Item mit der *ID* 25 in der Box mit der *BoxID* 17 von *Position* 8 auf *Position* 3. Dazu müssen zunächst alle Items in der Box an den Positionen 3 bis 7 um 1 nach hinten verschoben und anschließend muss der Attributwert in *Position* für das *Item* mit der *ID* 25 verändert werden.

(3 BE)

5 Neben den oben dargestellten Funktionen soll die App auch ermöglichen, dass Boxen neben Alben und Titeln auch Boxen enthalten dürfen. Dabei gilt, dass eine Box in maximal einer anderen Box enthalten sein darf. Ist sie in keiner Box enthalten, dann handelt es sich um eine sogenannte Root-Box.
Um dies zu gewährleisten wird das in Material 2 gegebene ER-Diagramm gemäß Material 3 verändert. Das ER-Diagramm aus Material 2 bleibt ansonsten unverändert.

5.1 Erläutern Sie die sich daraus ergebenden Veränderungen am Relationenmodell.

(1 BE)

5.2 Beschreiben Sie, wie sich für eine bestimmte Box ermitteln lässt, ob es sich um eine Root-Box handelt.
Implementieren Sie eine SQL-Anweisung, die die IDs und Bezeichnungen aller Root-Boxen ausgibt.

(3 BE)

5.3 Grundsätzlich soll es nicht möglich sein, dass sich Boxen wechselseitig enthalten, dass also z. B. die Box mit der ID 10 die Box mit der ID 11 enthält und die Box mit der ID 11 die Box mit der ID 10 enthält. Das Relationenmodell aus Aufgabe 3.2 verhindert dies jedoch nicht.

Geben Sie SQL-Anweisungen an, die zwei Boxen mit den IDs 10 und 11 erzeugen, die einander enthalten.

(3 BE)

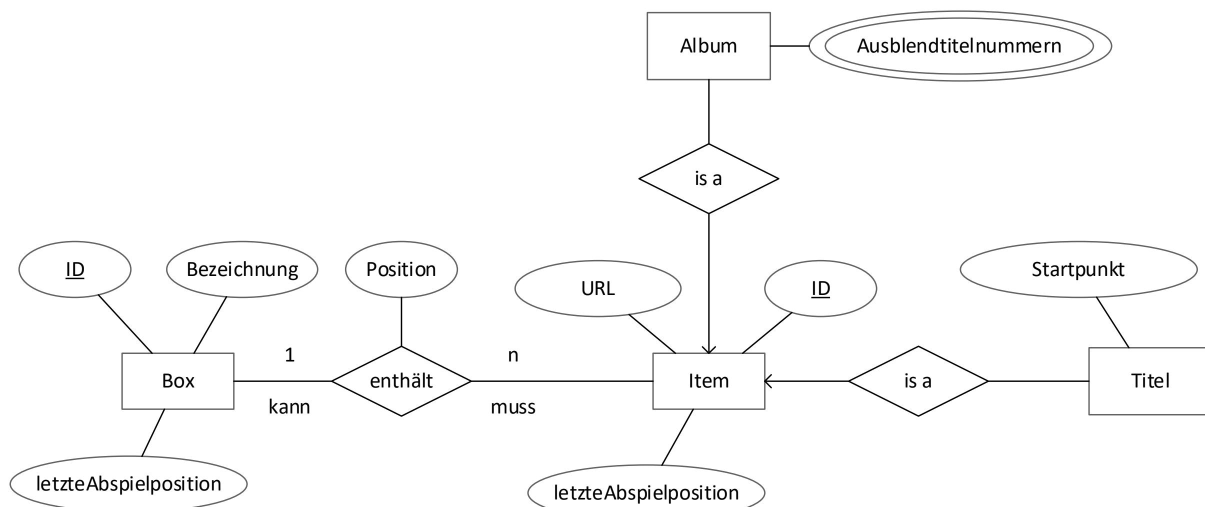
Material 1

Relation *Playlist*

<u>ID</u>	Name	Position und TitelID
5	Lovesongs	1) 5 2) 821
19	Chillout	1) 52 2) 821 3) 52
6	Lovesongs	1) 821

Material 2

ER-Diagramm



Material 3

Ausschnitt des veränderten ER-Diagramms

